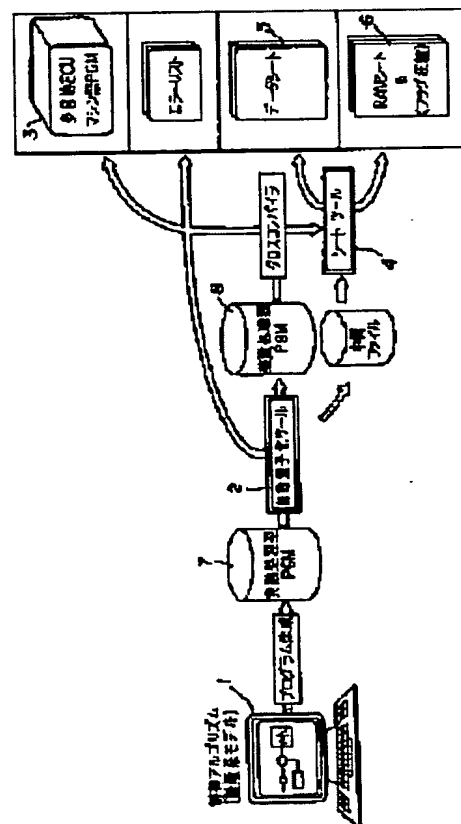


JP5334061

Patent number: JP5334061
Publication date: 1993-12-17
Inventor: MATSUI HIROKI; SAKURABA TOSHINORI; HIROTA TOSHIKI
Applicant: HONDA MOTOR CO LTD
Classification:
- international: G06F9/06; B60R16/02; F02D28/00; F02D45/00;
G06F7/38; G06F9/06; B60R16/02; F02D28/00;
F02D45/00; G06F7/38
- european:
Application number: JP19920164298 19920529
Priority number(s): JP19920164298 19920529

Abstract of JP5334061

PURPOSE: To improve the development efficiency of a control system by storing a second control program including a second formula described in a high-level language.
CONSTITUTION: A first control program described in the high-level language including a formula using real variables A and B is stored, and a first formula of $B = (\text{real constant } C) \times ((A/C) + \text{or } -0.5)$ is extracted from the first control program, and the first formula is converted to the second formula of $B = (A + \text{or } -\alpha) / \beta$ where $\alpha/\beta: 1/2$, and the second control program including the second formula described in the high-level language including the formula using real variables A and B is stored. Consequently, a real number processing-type control program (source program) 7 where the control algorithm is confirmed through simulation by a large-scale computer 1 is converted to an integer processing-type control program (object program) 8, and this object program 8 is converted to a machine language and is loaded to an ECU 3 on a vehicle.



Data supplied from the esp@cenet database - Worldwide

(19)日本国特許庁(JP)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平5-334061

(43)公開日 平成5年(1993)12月17日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	4 3 0 E	7232-5B		
	M	7232-5B		
B 6 0 R 16/02	M	2105-3D		
F 0 2 D 28/00	Z	7367-3G		
45/00	3 7 2 Z	7536-3G		

審査請求 未請求 請求項の数 2(全 14 頁) 最終頁に続く

(21)出願番号 特願平4-164298

(22)出願日 平成4年(1992)5月29日

(71)出願人 000005326

本田技研工業株式会社

東京都港区南青山二丁目1番1号

(72)発明者 松井 弘樹

埼玉県和光市中央1丁目4番1号 株式会社
本田技術研究所内

(72)発明者 桜庭 俊典

埼玉県和光市中央1丁目4番1号 株式会社
本田技術研究所内

(72)発明者 廣田 俊明

埼玉県和光市中央1丁目4番1号 株式会社
本田技術研究所内

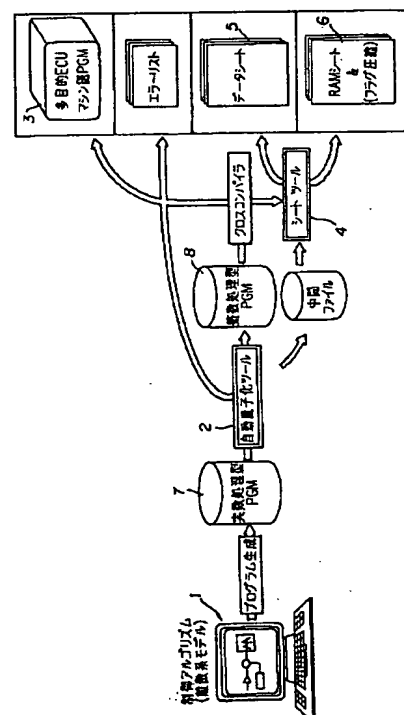
(74)代理人 弁理士 吉田 豊 (外1名)

(54)【発明の名称】 デジタル制御プログラムの生成変換装置

(57)【要約】

【構成】 大型コンピュータでシミュレーションを通じて生成した実数処理型のデジタル制御プログラム(ソースプログラム)を、整数処理型のオブジェクトプログラムに生成変換して車載ECUの低レベルのコンピュータにロードできる様にすると共に、数値の丸め誤差をシミュレーション結果と一致させる。

【効果】 人手を要することなく、シミュレーションの演算精度と同一なECUプログラムを自動生成することができる。



【特許請求の範囲】

【請求項1】 実数処理型の第1制御プログラムを整数処理型の第2制御プログラムに変換するデジタル制御プログラムの生成変換装置において、

a. 実数変数A、Bを用いた式を含む高級言語によって記述された第1制御プログラムを格納する第1格納手段、

b. 前記第1格納手段に格納された第1制御プログラムの中から

$$B = \text{実数定数} C \times ((A/C) \pm 0.5)$$

なる第1の式を抽出する手段、

c. 前記第1の式を

$$B = (A \pm \alpha) / \beta \quad (\text{但し、}\alpha/\beta : 1/2)$$

なる第2の式に変換する手段、

および

d. 実数変数A、Bを用いた式を含む高級言語によって記述された前記第2の式を含む第2制御プログラムを格納する第2格納手段、を具備することを特徴とするデジタル制御プログラム生成変換装置。

【請求項2】 請求項1項において生成された前記第2の式を備えたことを特徴とする車載電子制御ユニット。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 この発明はデジタル制御プログラムの生成変換装置に関し、より具体的には大型コンピュータでシミュレーションを通じてアルゴリズムを検証したデジタル制御プログラムを、車両などに搭載する低レベルのコンピュータにそのままロードできる様にしたものに関する。

【0002】

【従来の技術】 従来、マイクロ・コンピュータを用いた制御システム、例えば車載電子制御ユニット（以下「ECU」と称する）を用いて車両運転制御システムを設計するときは、技術者が仕様書を作成した後、それに基づいて大勢のプログラマがプログラムを作成していた。その結果、開発に多大の人手と工数が必要となると共に人為ミスも発生していた。また近時、特開平1-171066号公報で提案される様に、大型コンピュータを使用してシミュレーションを通じて制御システムを開発することも提案されているが、その場合も事情は同様であった。即ち、シミュレーション用のコンピュータは記憶容量も大きいことから、浮動少数点形式を用いて演算がなされるが、実機に搭載されるECUは記憶容量が小さく、また実機では制御周期が燃焼サイクルごととなって短時間であることから、浮動少数点形式を用いて演算を行っていたのでは、制御が間に合わなかった。そのため、大型コンピュータでシミュレーションを行って制御アルゴリズムを検証した後、ECUのプログラムの作成は依然として人手に頼らざるを得なかった。

【0003】

【発明が解決しようとする課題】 そこで本出願人は先に特願平2-336,424号（平成2年11月30日出願）において、大型コンピュータで制御アルゴリズムを検証して生成した実数処理型の制御プログラム（ソースプログラム）から、そのままECUにロードできる整数処理型の制御プログラム（オブジェクトプログラム）を生成変換する技術を提案している。

【0004】 ところで、コンピュータの数値演算においては実数の丸め誤差が必然的に伴うが、前記の先に提案した技術においてもソースプログラムとオブジェクトプログラムとの間に丸め誤差が生じていた。一般的には、自然法則によって支配される物理現象理論と、大型コンピュータで計算シミュレートした制御結果と、低レベルのコンピュータ（ECU）を用いて実際に制御した結果の三者は、互いに少ない誤差で近似されることが望ましい。特に、大型コンピュータにおける計算シミュレート結果と実際の制御結果の一致は、制御系の開発効率を考える上で重要な問題である。

【0005】 従って、この発明の目的は上記した欠点を解消し、大型コンピュータ上で記述されたデジタル制御プログラム（ソースプログラム）からECUにロード自在なオブジェクトプログラムを自動的に生成変換すると共に、その間に数値の丸め誤差が生じない様にしたデジタル制御プログラムの生成変換装置を提供することを目的とする。

【0006】

【課題を解決するための手段】 上記の目的を解決するために本発明は例えば請求項1項において、実数処理型の第1制御プログラムを整数処理型の第2制御プログラムに変換するデジタル制御プログラムの生成変換装置において、実数変数A、Bを用いた式を含む高級言語によって記述された第1制御プログラムを格納する第1格納手段、前記第1格納手段に格納された第1制御プログラムの中から

$$B = \text{実数定数} C \times ((A/C) \pm 0.5)$$

なる第1の式を抽出する手段、前記第1の式を

$$B = (A \pm \alpha) / \beta \quad (\text{但し、}\alpha/\beta : 1/2)$$

なる第2の式に変換する変換手段、および実数変数A、Bを用いた式を含む高級言語によって記述された前記第2の式を含む第2制御プログラムを格納する第2格納手段、を具備する如く構成した。

【0007】

【作用】 シミュレーションとECU搭載用プログラムの演算精度を一致させることができるので、制御システムの開発効率を大幅に向上させることができると共に、自然法則で支配される物理現象理論を含めた三者の一致に一步前進させることができる。

【0008】

【実施例】 図1は開発システムを全体的に示す説明図であり、大型コンピュータ（ワークステーション）1にお

いて構築した離散系モデルに対してシミュレーションを行って、車両運転のデジタル制御プログラム（ソースプログラム）が開発される。これは浮動少数点形式を用いた実数処理型のプログラムである。そして、図中に「自動量子化ツール」2として示すのがこの発明に係るデジタル制御プログラムの生成変換装置であり、大型コンピュータ1で生成された実数処理型プログラムを読み込んで後で述べる様に整数処理型プログラムに生成変換する。生成された整数処理型プログラムは公知のクロスコンパイラによってマシン語に変換され、車両に搭載されたマイクロ・コンピュータからなる電子制御ユニット（ECU）3に、RS232Cなどの送信方式やROMなどを媒介としてロードすることができる。量子化ツール2は大型コンピュータ1と同程度の記憶容量を備えるコンピュータからなる。尚、この発明に係る自動量子化ツール2を介して生成変換される整数処理型プログラム（オブジェクトプログラム）に基づき、第2のツール（図中にシートツール4として示す）を介して第2、第3のプログラム（図中にデータシート5、ラムシート6として示す）が生成されるが、それらはこの発明の要旨とは関係ないので、説明は省略する。

【0009】図2に自動量子化ツール2の構成を機能的に示す。

【0010】以下、図3フロー・チャート以降を参照してこの発明に係るデジタル制御プログラムの生成変換装置（自動量子化ツール2）の動作を説明する。

【0011】図3はそのメイン・ルーチン・フロー・チャートであり、同図に従って説明すると、先ずS10においてソースプログラムから量子化情報を検出する。

「量子化」は一般にデジタル制御において離散値に丸めることを意味するが、この明細書でも同様に数値を丸める意味で使用している。また、この明細書では量子化（丸め）処理が必要な情報を「量子化情報」と呼び、このステップで検出する。

【0012】図4はその作業を示すサブルーチン・フロー・チャートであり、先ずS100においてソースプログラムのファイルを読み込む。即ち、実数処理型制御プログラムの一連のステートメントの1つを読み込む。この一連のステートメントの1つは「ブロック」と呼ばれる。

【0013】次いでS102に進んで最終ファイルではないことを確認し、S104に進んで読み込んだブロックのキーを検索する。各ブロックにはそれぞれ多項式演算、論理演算などその内容を示すキーが付されており、うち前記した量子化情報ブロックには、予め「Quantizer」なるキーが付されているものとする。従って、S106に進んで読み込んだブロックが「Quantizer」であるか否か判断し、肯定されるときS108以降に進むと共に、否定されたときは直ちにプログラムを終了する。

【0014】フロー・チャートの説明を続ける前に、こ

こでこの発明に係る制御プログラムの生成変換技術を概説する。

【0015】先にも述べた様に、シミュレーション用の大型コンピュータの制御プログラムを低レベルのECU用の制御プログラムに変換しようとするとき、支障となるのが、数値の処理である。車両制御の場合には制御周期の制約があることから、現在ECUに多用されている低レベルのコンピュータで浮動少数点演算を行うのは困難である。そして、数値処理の問題はスケール合わせ、即ち、スロットル開度などの制御パラメータについて搭載を予定するECUでセンサ出力をA/D変換する回路の分解能に応じて再設定する際など頻繁に生じる。そこで、本出願人が先に提案した技術では、実数を分数化すれば整数化できることに着目して実数処理プログラムから整数処理プログラムに生成変換することを実現した。

【0016】その先の技術で具体的には、制御プログラム（ソースプログラム）において例えば上に述べたスケール合わせに関して言えば、

実数変数B = 実数定数C × 丸め関数（実数変数A / 実数定数C）

なる表現を行っておいた（上式で実数定数CはECUの1LSB当たりのセンサ分解能、例えば0.098（スロットル開度 = 1LSB）である）。次いで、プログラム生成変換装置において上式などの量子化処理が必要な表現を量子化情報として検出し、実数定数C、例えば0.098を98/1000と分数で一旦表現して整数系での処理を可能とした上で、

実数変数B = 実数変数A

とそのまま置換していた。

【0017】即ち、先に提案した技術では、丸め関数として切り捨て関数を用いていたため、シミュレーション上では丸め関数として4捨5入方式を用いて量子化処理を行うと、先に提案した技術の量子化処理結果は、シミュレーション結果と一致しない不都合があった。そこで、この発明においてはその点を改良し、先ずソースプログラムに

実数変数B = 実数定数C × （（実数変数A / 実数定数C） ± 0.5）

なる表現を行っておくと共に、先に示した分数による整数化の概念に従い、この式を置換してシミュレーション結果と一致させる様にした。具体的には入力側（横軸）の精度を2倍に上げ、次いで正負に応じて1LSB（2倍に上げたことから0.5）を加減算し、次いで出力側（縦軸）の精度自体はECU側のセンサ分解能で規定されて変更しようがないことから、それに対応させるために2で除算する。その結果、シミュレーション結果との一致を見たものである。具体的には、プログラム生成変換装置において、

実数変数B = （実数A ± 1） / 2

と置換する様にした。換言すれば、4捨5入関数0.5

が本来的には実数表現であって、整数処理にはなじまないものであるが、これも分数表現して整数処理を可能とした。

```
if (BUS [1] >= 0)
    BUS [2] = 0.098 × int (BUS [1] / 0.098 + 0.5)
else
    BUS [2] = 0.098 × int (BUS [1] / 0.098 - 0.5)
```

などと表現される。ここで、BUS は変数名（より正確にはその配列）を意味し、BUS [1] を入力BUS、BUS [2] を出力BUS とする（入力BUS の引数 [1] を k 、出力BUS のそれ [2] を l で示す）。また、0.098 は前述の如く ECU 側の 1 LSB 当たりのセンサ分解能を意味し、先の提案で述べたと同様に $98/1000$ と整分数化する。この明細書ではその分子を量子化係数（最初は Quantize 値と仮称する）と称し、その逆数を量子化倍率と称する。図5はその分数を記憶する量子化倍率表（テーブル）である。尚、int は丸め関数を意味する。

【0019】以上を前提として図4フロー・チャートに戻って説明を続けると、S106でQuantizer ブロックであることが確認されるとS108に進み、入出力BUSの引数 k 、 l の値を検索し、S110でQuantize 値（例で言えば、98）を検索し、S112に進んで先に述べた理由から入力BUS（例で言えば $k=1$ ）の量子化係数をQuantize 値の $1/2$ に設定し、次いでS114に進んで出力BUS（ $l=2$ ）の量子化係数としてQuantize 値をそのまま設定する。以上の処理を、S102で最終ファイルと判断されるまで繰り返す。

【0020】図3フロー・チャートに戻ると、次いでS12に進んで前記した量子化倍率を決定する。即ち、Quantizer ブロックについてはS10で量子化係数（倍率）を検出（決定）し終わっているが、その他の代数、論理、テーブル検索などの実数については未だ決定していないため、ここで決定する。

【0021】図6はその作業を示すサブルーチン・フロー・チャートであり、先ずS200において先のS100と同様にソースプログラムのファイルを読み込み、S202に進んでブロックのキーを検索し、S204に進んで入力側の量子化係数が設定されているか否か判断し、設定されるまでS200からS204を繰り返すと共に、設定されているときはS206に進んでブロックの種類に応じて出力側の量子化係数を設定する。入力側と出力側についての量子化係数（倍率）の設定手法は、図4フロー・チャートで説明したのと異ならない。

【0022】次いでS208に進んでソースプログラムの終わりまで処理したか否か判断し、否定されるときはS200～S208を繰り返すと共に、S208で肯定されるときはS210に進んで量子化係数が図5に示すテーブルで全て書き込まれたか否か判断し（既述の如く図5テーブルで分子が量子化係数を意味する。量子化倍

【0018】尚、上に述べたソースプログラムでの量子化情報は実施例ではC言語を用いていて、

率はその逆数である）、否定されるときはS212に進んでソースプログラムの先頭に戻って以上の処理を繰り返す。

【0023】図3フロー・チャートに戻ると、次いでS14に進んでオブジェクトプログラムを生成する。

【0024】図7はそれを示すサブルーチン・フロー・チャートであり、先ずS300においてソースプログラムのファイルを読み込み、S302に進んで最終ファイルではないことを確認してS304に進んでブロックのキーを再び検索し、S306に進んで検索したブロックの種類を判断し、演算ブロックであればS308に進んで演算式構文解析木処理を行う。

【0025】図8はそれを示すサブルーチン・フロー・チャートであり、先ずS400においてトークンの取り出しとトークンテーブルへの登録を行う。ここでトークンとは鍵となる文字を区切りとする1つの単語を意味する。具体的には、

BUS [1] = BUS [2] × BUS [3] × 4.0 ;

なる文があるとき、BUS [1]、=、BUS [2]、×、BUS [3]、×、4.0、;、のそれぞれをトークンと称する。従って、この文は8つのトークンから構成される。

【0026】S400では読み込んだ演算ブロックについて、先ず『=』までのトークンを取り出し、トークンテーブルに登録する。次いでS402に進んで次のトークン、上の例で言えば『BUS [2]』、を取り出す。次いでS404に進んでトークンの種類が実数か変数か判断する。予め、文字で始まるトークンは変数、数字で始まるトークンは実数と認識する様に約束しておく。そして、上の例で言えば変数と判断されるので、S406に進んで1つ前のトークンの演算子の種類を判断する。この様にしたのは、代数演算であれば、変数（実数）と演算子とが交互する筈だからである。上の例でいえば

『=』であるので、S408に進んでトークンテーブルに登録すると共に、量子化倍率への追加を行う。この量子化倍率の追加は、変数にはなんらかの係数が設定されているが、量子化倍率を乗じられた整数演算として入ってくるので、それを使用して後で述べる様に代数演算を行うときは、その倍率で除算して一旦もとの値に戻す必要があることから、その処理のためである。

【0027】またS406で乗算か除算と判断されるときはS410に進んで2つ前のトークンの種類を判断し、変数であればS408に進むと共に、実数であれば

S 4 1 2 に進んで、トークンテーブルの入れ換えを行う。これは実数が 2 つ以上あって演算が可能ならば演算を行うためと、除算があるときは桁落ちを防止するために最後に行う様にするためである。また先の S 4 0 4 でトークンが実数であると判断されるときは S 4 1 6 に進んで 1 つ前のトークンの演算子を判断し、S 4 1 8 から S 4 2 2 で同様の処理を行う。

【0028】次いで S 4 2 4 に進んで、先に述べた様に代数演算であれば実数（変数）の後は演算子か代入文しかないことから、直ちに次のトークンの取り出しとトークンテーブルへの登録を行う様にして（即ち、S 4 0 0 への復帰を省略して）処理の簡略化を図り、次いで S 4 2 6 に進んで文末の『:』に達したと判断されない限り、S 4 0 2 に戻って以上のループを繰り返す。

【0029】図 7 フロー・チャートに戻ると、次いで S 3 1 0 に進んでトークンテーブルのテキストプログラム化を行う。これは図 8 フロー・チャートで得られたものを、そのままコード生成することを意味する。但し、実数については整分数化（約分含む）する。

【0030】また、図 7 フロー・チャートにおいて S 3 0 6 で Quantizer ブロックと判断されたときは S 3 1 2 に進んで Quantizer のテキストプログラム化を行う。

【0031】その作業を示す図 9 サブルーチン・フロー・チャートを参照して説明すると、先ず S 7 0 0 で入出力 BUS の引数 k, l （即ち、BUS の [] 内の数値）を検索し、次いで S 7 0 2 に進んで図示のフォーマットに基づいてコード生成する。この内容については既に述べた。

【0032】また、図 7 フロー・チャートにおいて S 3 0 6 でテーブル検索と判断されるときは S 3 1 4 に進んで、テーブル検索のテキストプログラム化を行う。

【0033】その作業を示す図 10 サブルーチン・フロー・チャートを参照して説明すると、S 8 0 0 でブロックのキーを再度読み込み、S 8 0 2 に進んでテーブルの種類が 2 次元テーブル（『2D』と略記）か 3 次元テーブルか否か判断する（即ち、ブロックのキーには『2D LOOKUP TABLE』『3D LOOKUP TABLE』などのキーが予め付されている）。そして S 8 0 2 で 2 次元テーブルと判断されるときは S 8 0 4 に進んで所定のフォーマットでコード生成してテキストを生成する（具体的には引数の値を読み込んで代入して行う）と共に、S 8 0 2 で否定されたときは 3 次元テーブルとなるので、S 8 0 6 に進んで演算の順序を考慮してテキストを生成する。これは 3 次元テーブルの場合には 2 次元のそれに比して構成が複雑となって演算が入ることから、代数演算で述べたと同じ理由で除算を最後にするなどの処理を行ってテキストを生成する。最後に S 8 0 8 に進んで零割りなどの異常がないか否かチェックして終わる。

【0034】また、図 7 フロー・チャートにおいて、S 3 0 6 で論理演算と判断されるときは S 3 1 6 に進ん

で、論理式の構文解析木処理を行う。図 11 の S 9 0 0 から S 9 2 6 にその処理を示す。論理演算の場合には、If (BUS [1] > 0) . . .

などと表現される点を除き、その構文解析木処理は図 8 で述べた代数演算の場合と殆ど同一なので、説明は省略する。続いて、図 7 フロー・チャートの戻って S 3 1 8 に進み、トークンテーブルのテキストプログラム化を行う。これも代数演算と同様である。

【0035】続いて、再び図 3 に戻り、最終の S 1 6 に進んでオブジェクトプログラムとしてのテキストプログラムを出力して終わる。

【0036】この発明に係る生成変換装置（自動量子化ツール 2）で以上の処理を行うことにより、図 1 に示す如く、大型コンピュータ 1 でシミュレーションを通じて制御アルゴリズムを確認した実数処理型の制御プログラム（ソースプログラム）7 を、整数処理型の制御プログラム（オブジェクトプログラム）8 に生成変換することができる。生成変換されたオブジェクトプログラムは公知のクロスコンパイラを通じてマシン語に変換されて車載 ECU 3 にロードされ、所望の車両制御を行うことができる。

【0037】また、生成変換されたオブジェクトプログラムにおいては、量子化処理（数値の丸め処理）が大型コンピュータでのシミュレーション結果と一致しているので、制御システムの開発効率を大幅に向上させることができる。更には、初めに述べた自然法則によって支配される物理現象理論とシミュレーション結果と ECU での制御結果の三者の一致に一歩前進することができた。また ECU での実際の制御を大型コンピュータでシミュレーションを通じて検証するときも、両者の演算精度（量子化処理）が同一なため、容易に大型コンピュータにロードすることができる。即ち、両者に演算精度の不一致がないため、演算誤差をシミュレートするための演算を新たに大型コンピュータ上で付加する必要がない。

【0038】

【発明の効果】請求項 1 項は、実数処理型の第 1 制御プログラムを整数処理型の第 2 制御プログラムに変換するデジタル制御プログラムの生成変換装置において、実数変数 A, B を用いた式を含む高級言語によって記述された第 1 制御プログラムを格納する第 1 格納手段、前記第 1 格納手段に格納された第 1 制御プログラムの中から $B = \text{実数定数 } C \times ((A/C) \pm 0.5)$

なる第 1 の式を抽出する手段、前記第 1 の式を

$$B = (A \pm \alpha) / \beta \quad (\text{但し、} \alpha / \beta : 1/2)$$

なる第 2 の式に変換する手段、および実数変数 A, B を用いた式を含む高級言語によって記述された前記第 2 の式を含む第 2 制御プログラムを格納する第 2 格納手段を具備する如く構成したので、自然法則によって支配される物理現象理論と大型コンピュータで計算シミュレートした制御結果と低レベルのコンピュータを用いて実際に

制御した結果の三者の誤差をより少なくできると共に、シミュレーション結果と低レベルのコンピュータでの結果の演算精度を同一にしたので、制御システムの開発効率を大幅に向上させることができる。

【0039】請求項2項の車載電子制御ユニット、請求項1項において生成された前記第2の式を備える様に構成したので、請求項1項で述べた効果に加えて、センサ出力を忠実に反映する値を入力することができ、シミュレーションで検証した制御アルゴリズム通りの制御精度を実現することができる。

【図面の簡単な説明】

【図1】この発明に係るデジタル制御プログラムの生成変換装置（量子化ツール）を含む制御プログラムの生成過程を示す概略図である。

【図2】図1中のデジタル制御プログラムの生成変換装置（量子化ツール）の構成を示す機能ブロック図である。

【図3】この発明に係るデジタル制御プログラムの生成変換装置の動作を示すメイン・ルーチン・フロー・チャートである。

【図4】図3フロー・チャートの中の量子化情報の検出作業を示すサブルーチン・フロー・チャートである。

【図5】この発明に係るデジタル制御プログラムの生成変換装置で使用する量子化倍率表（テーブル）を示す

説明図である。

【図6】図3フロー・チャートの中の量子化倍率の決定作業を示すサブルーチン・フロー・チャートである。

【図7】図3フロー・チャートの中のオブジェクトプログラム生成作業を示すサブルーチン・フロー・チャートである。

【図8】図7フロー・チャートの中の演算式構文解析木処理作業を示すサブルーチン・フロー・チャートである。

【図9】図7フロー・チャートの中の量子化情報（Quantizer）テキストプログラム化作業を示すサブルーチン・フロー・チャートである。

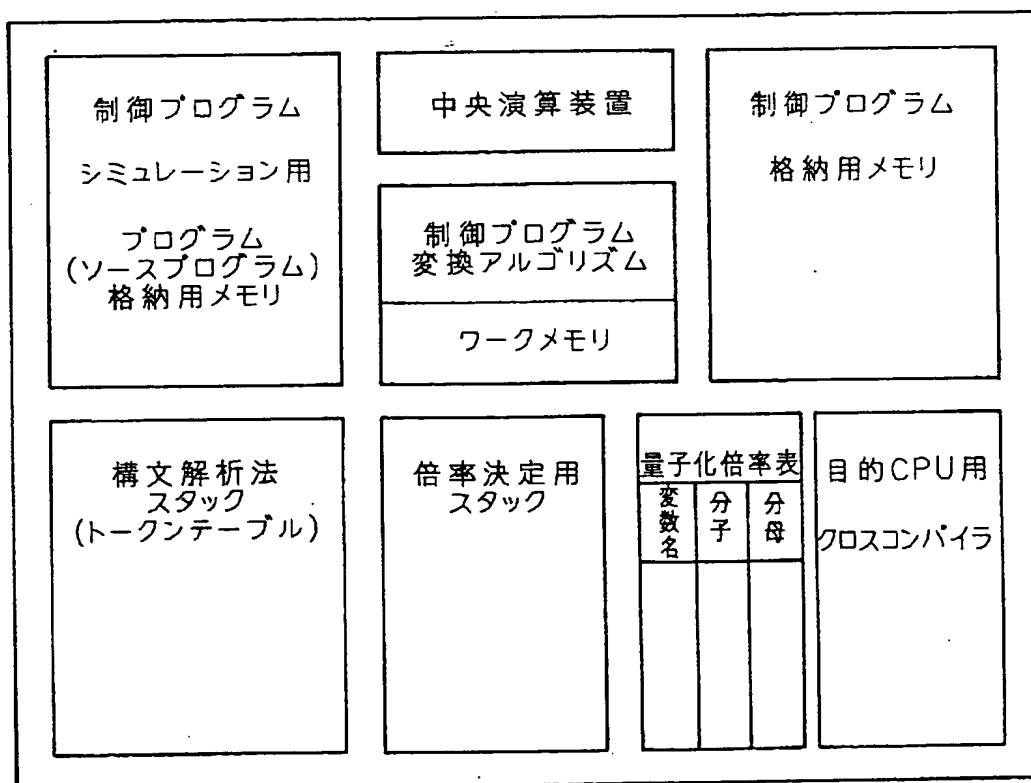
【図10】図7フロー・チャートの中のテーブル検索のテキストプログラム化作業を示すサブルーチン・フロー・チャートである。

【図11】図7フロー・チャートの中の論理式構文解析木処理作業を示すサブルーチン・フロー・チャートである。

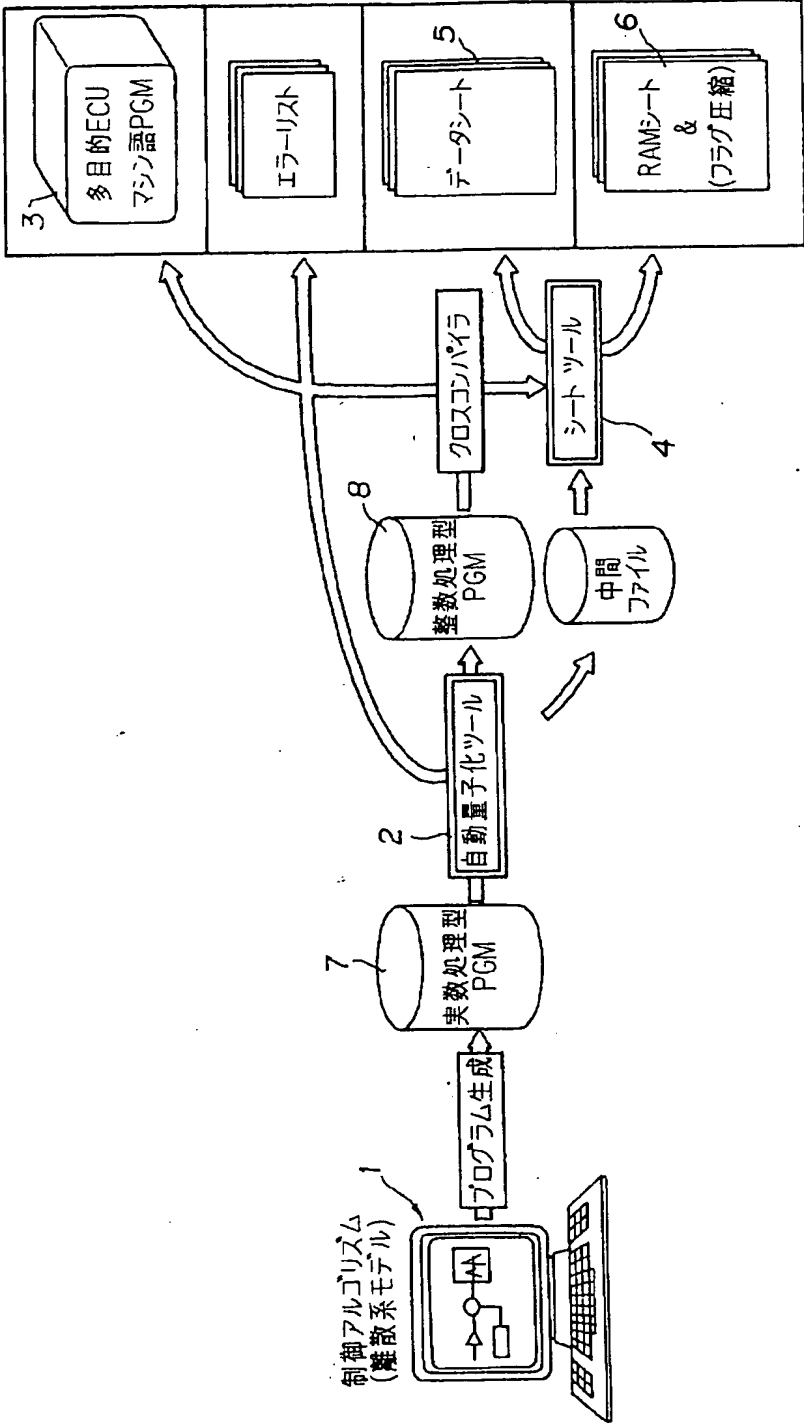
【符号の説明】

- 1 大型コンピュータ（シミュレーション用）
- 2 デジタル制御プログラム生成変換装置（自動量子化ツール）
- 3 車載電子制御ユニット（ECU）

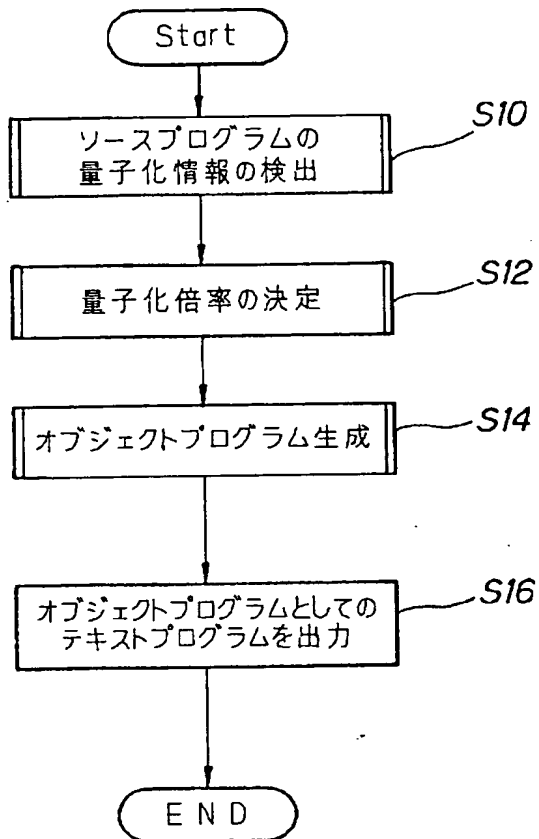
【図2】



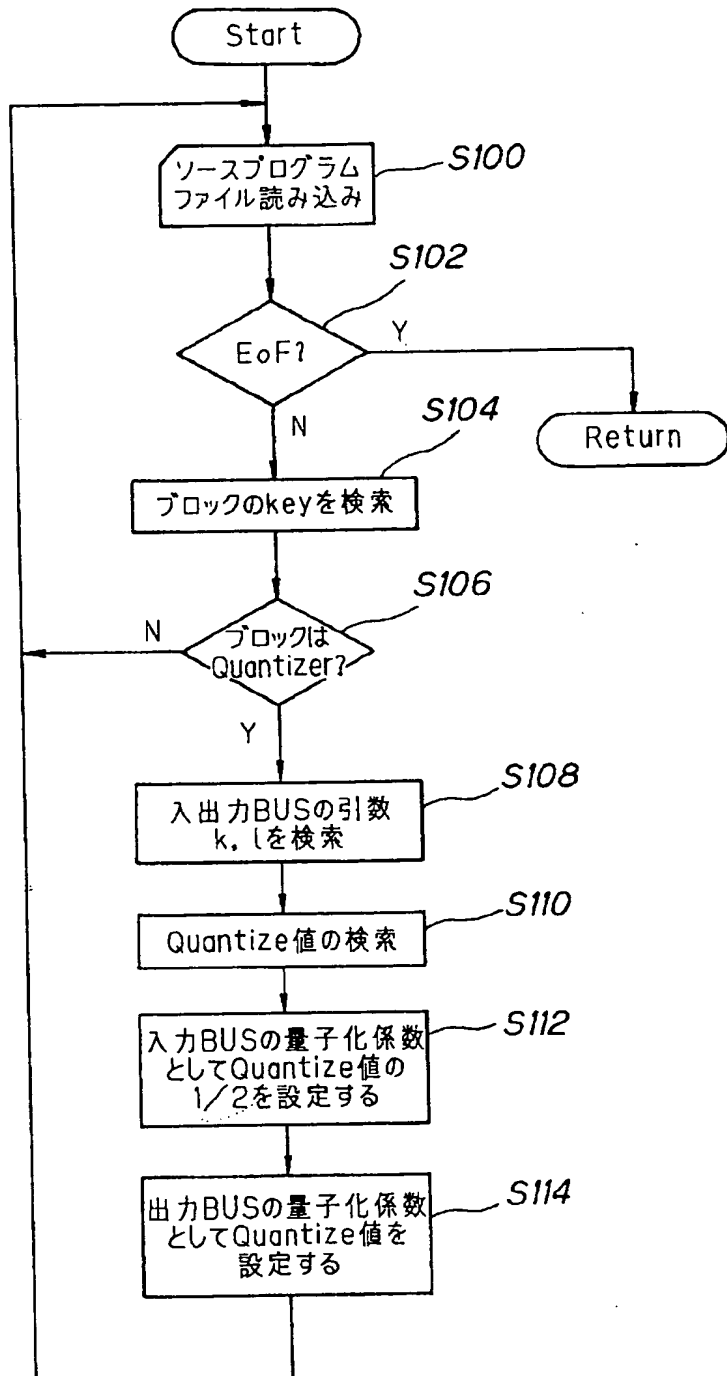
【図1】



【図3】



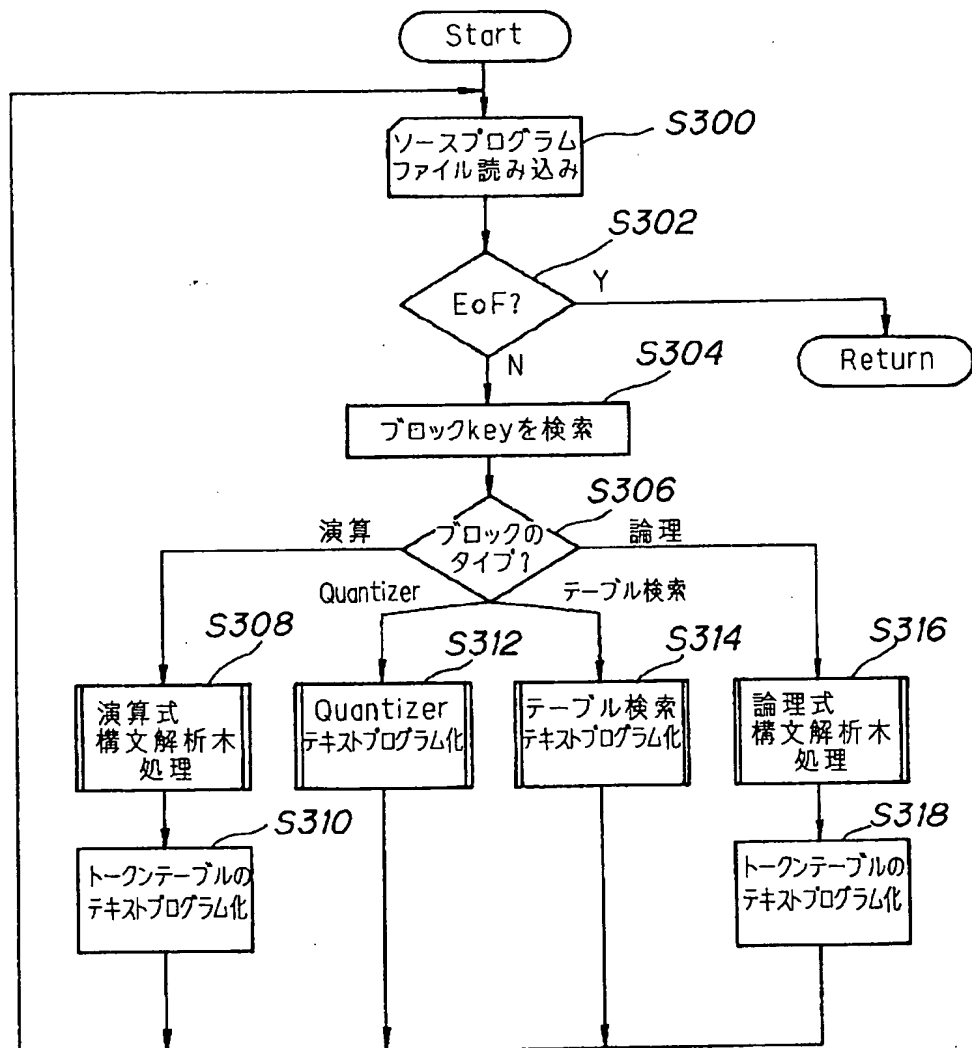
【図4】



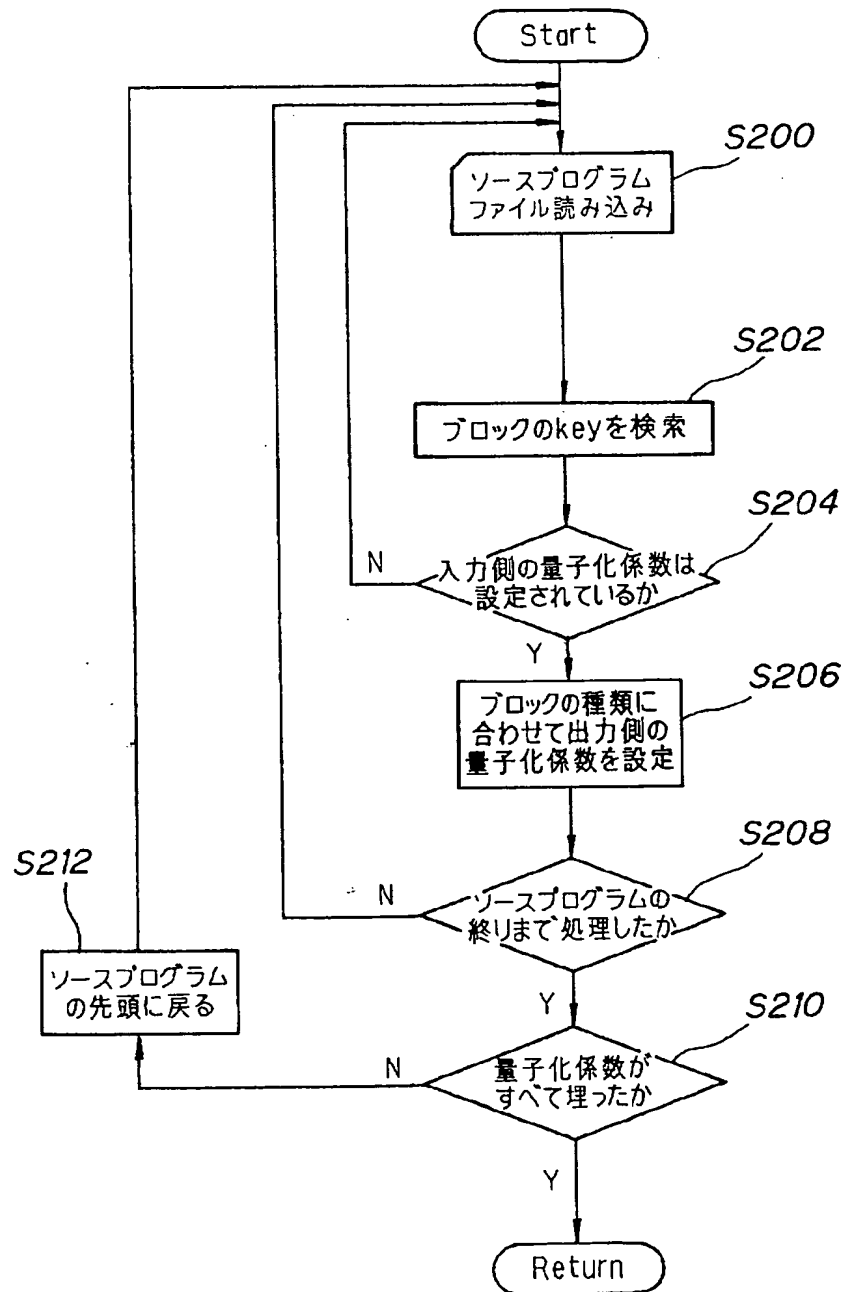
【図5】

変数名	分子	分母
A	98	1000
B		
⋮		

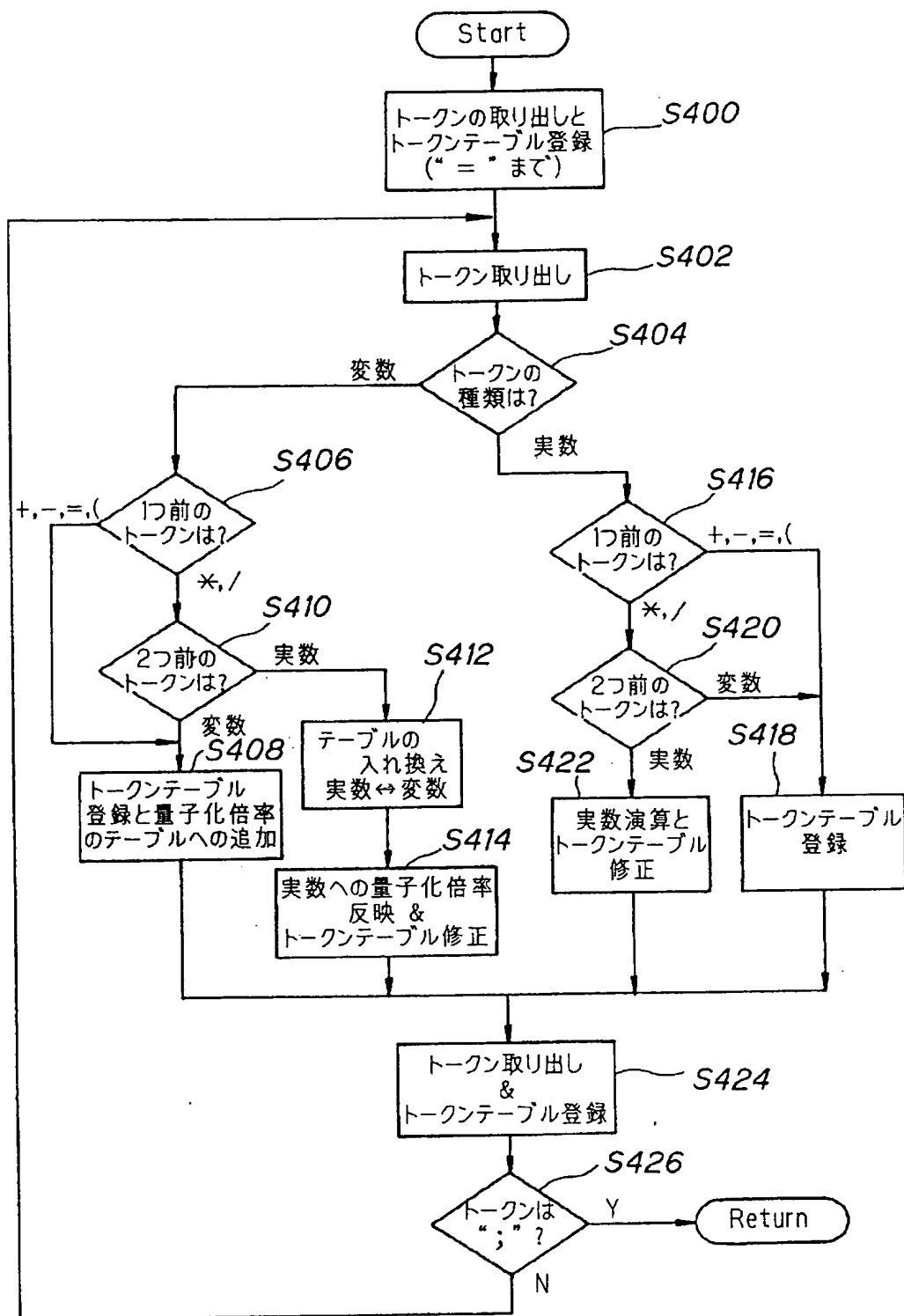
【図7】



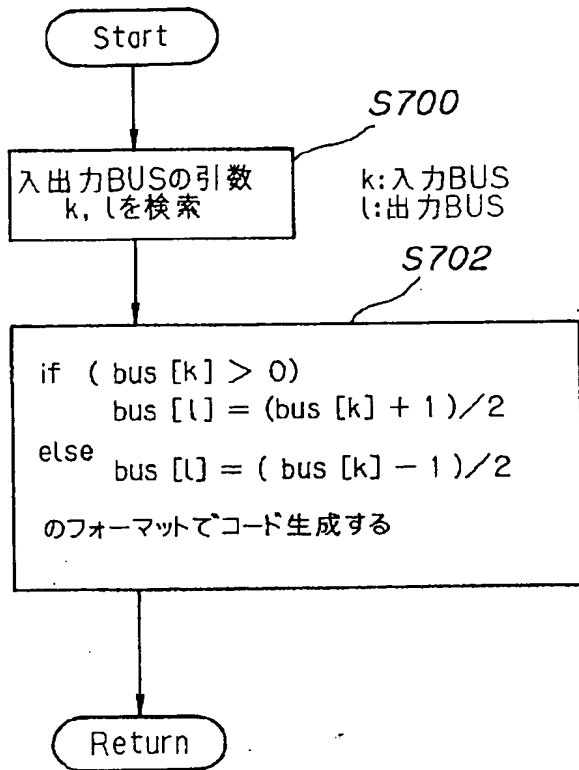
【図6】



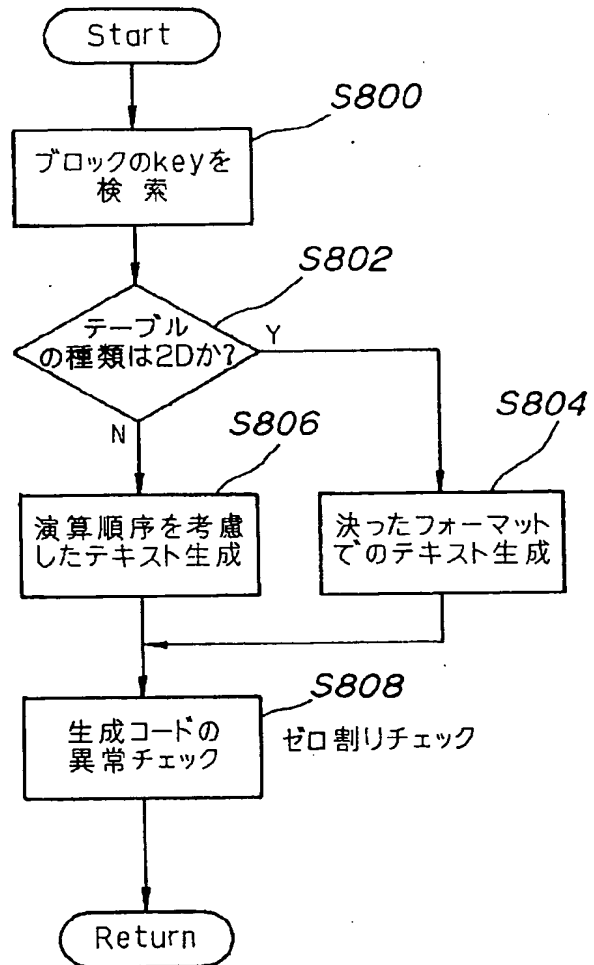
【図8】



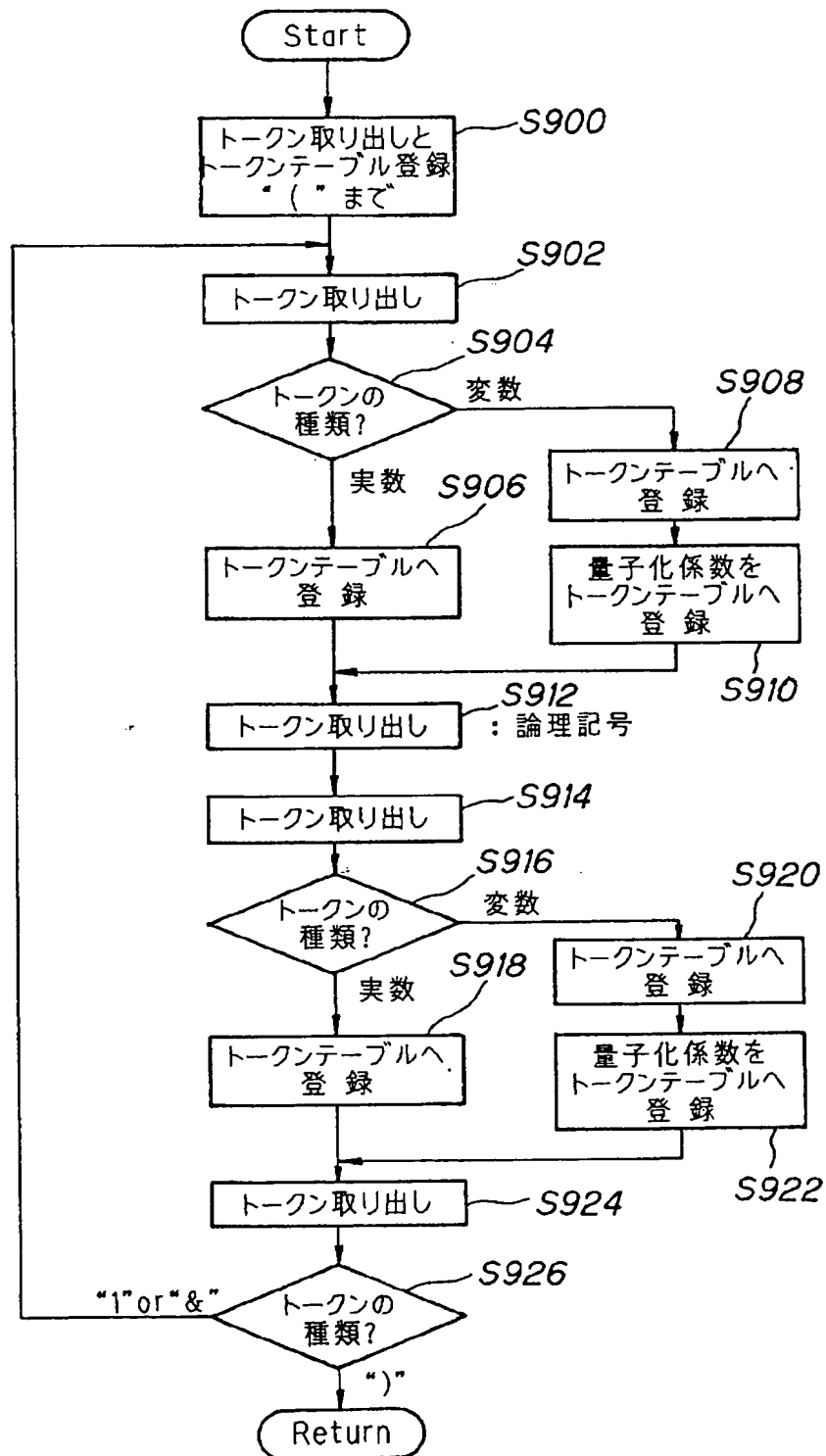
【図9】



【図10】



【図11】



フロントページの続き

(51) Int. Cl. 5

G 0 6 F 7/38

識別記号

庁内整理番号

F I

技術表示箇所

B 9291-5B